

AMENDMENTS TO THE CLAIMS

Please amend the claims as follows.

1. (Currently Amended) A method for instantiating ~~an object~~ a first function, comprising:
~~determining an object type of said object;~~
reserving a memory block on a memory structure, the size of said memory block being
determined according to said ~~object type~~ first function, and said memory structure
being selected according to said ~~object type~~ first function; and
creating a reference structure to said ~~object~~ first function,
wherein ~~the object~~ said first function is written in a dynamically typed language[[]], and
wherein said first function comprises a definition of a second function.
2. (Cancelled)
3. (Currently Amended) The method of claim [[2]] 1, further comprising:
executing a set of constructor statements if said set contains at least one statement.
4. (Cancelled)
5. (Currently Amended) The method of claim [[4]] 1, wherein said memory structure is a heap.
6. (Cancelled)
7. (Currently Amended) The method of claim [[6]] 1, further comprising:
optionally returning a value to a calling statement;
deleting said reference structure; and
freeing said memory block.
8. (Currently Amended) The method of claim [[7]] 1, wherein said memory structure is a stack.

9. (Currently Amended) A computer program product comprising:

a computer usable medium having computer readable program code embodied therein configured to instantiate ~~an object~~ a first function, said computer program comprising:

~~computer readable code configured to cause a computer to~~

~~determine an object type of said object;~~

computer readable code configured to cause a computer to

reserve a memory block on a memory structure, the

size of said memory block being determined according to said ~~object~~

~~type~~ first function, and said memory structure being selected

according to said ~~object type~~ first function; and

computer readable code configured to cause a computer to

create a reference structure to said ~~object~~ first function,

wherein ~~the object~~ said first function is written in a dynamically typed language[[.]], and

wherein said first function comprises a definition of a second function.

10. (Cancelled)

11. (Currently Amended) The computer program product of claim [[10]] 9, further comprising:

computer readable code configured to cause a computer to execute a set of constructor statements if said set contains at least one statement.

12. (Cancelled)

13. (Currently Amended) The computer program product of claim [[12]] 9, wherein said memory structure is a heap.

14. (Cancelled)

15. (Currently Amended) The computer program product of claim [[14]] 9, further comprising:
computer readable code configured to cause a computer to optionally return a value to a calling statement;
computer readable code configured to cause a computer to delete said reference structure;
and
computer readable code configured to cause a computer to free said memory block.
16. (Currently Amended) The computer program product of claim [[15]] 9, wherein said memory structure is a stack.
17. (Currently Amended) A system for instantiating ~~an object~~ a first function comprising:
~~an interpreter configured so as to differentiate object types;~~
a storage allocation subsystem configured so as to reserve a storage block on a memory device, said allocation subsystem further configured to select the size of said storage block and said memory device according to said ~~object type~~ first function; and
an access control subsystem, said access control subsystem creating a reference structure for said object,
wherein ~~the object~~ said first function is written in a dynamically typed language[[.]], and
wherein said first function comprises a definition of a second function.
18. (Cancelled)
19. (Currently Amended) The system of claim [[18]] 17, further comprising:
a statement execution subsystem, said execution subsystem configured so as to automatically execute a set of constructor statements.
20. (Cancelled)
21. (Currently Amended) The system of claim [[20]] 17, wherein said memory device is a heap.
22. (Cancelled)

23. (Currently Amended) The system of claim ~~[[22]]~~ 19, wherein said execution subsystem is further configured so as to:
- optionally return a value to an object calling subsystem.
24. (Original) The system of claim 23, wherein said storage allocation subsystem is further configured to automatically delete said reference structure and automatically freeing said storage block after said statement execution subsystem completes the execution of said set of constructor statements and completes the optional return of a value to said object calling subsystem.
25. (Currently Amended) The system of claim ~~[[24]]~~ 17, wherein said memory device is the stack.
26. (Currently Amended) ~~An object~~ A first function instantiation component for an operating system, comprising:
- ~~an interpreter configured so as to differentiate object types;~~
 - a storage allocation subsystem configured so as to reserve a storage block on a memory device, said allocation subsystem further configured to select the size of said storage block and said memory device according to said ~~object type~~ first function; and
 - an access control subsystem, said access control subsystem creating a reference structure for said object,
- wherein ~~the object~~ said first function is written in a dynamically typed language~~[[.]], and~~
wherein said first function comprises a definition of a second function.
27. (Cancelled)
28. (Currently Amended) The component of claim ~~[[27]]~~ 26, further comprising:
- a statement execution subsystem, said execution subsystem configured so as to automatically execute a set of constructor statements.
29. (Cancelled)
30. (Currently Amended) The component of claim ~~[[29]]~~ 26, wherein said memory device is a heap.
31. (Cancelled)

32. (Currently Amended) The component of claim [[31]] 28, wherein said execution subsystem is further configured so as to:
optionally return a value to an object calling subsystem.
33. (Previously Presented) The component of claim 32, wherein said storage allocation subsystem is further configured to automatically delete said reference structure and automatically freeing said storage block after said statement execution subsystem completes the execution of said set of constructor statements and completes the optional return of a value to said object calling subsystem.
34. (Currently Amended) The component of claim [[33]] 26, wherein said memory device is a stack.
35. (Cancelled)
36. (Cancelled)
37. (Cancelled)
38. (Currently Amended) A method for instantiating ~~an object~~ a first function, comprising:
~~determining an object type of said object;~~
reserving a memory block on a memory structure, the size of said memory block being determined according to said ~~object type~~ first function, and said memory structure being selected according to said ~~object type~~ first function;
creating a reference structure to said ~~object~~ first function; and
~~wherein the object includes a definition of one or more objects of the object type, and~~
wherein ~~the object~~ said first function is written in a dynamically typed language[[.]], and
wherein said first function comprises a definition of a second function.
39. (Cancelled)
40. (Currently Amended) The method of claim [[39]] 38, further comprising:
executing a set of constructor statements if said set contains at least one statement.
41. (Cancelled)

42. (Currently Amended) The method of claim ~~[[41]]~~ 38, wherein said memory structure is a heap.

43. (Cancelled)

44. (Currently Amended) The method of claim ~~[[43]]~~ 38, further comprising:

optionally returning a value to a calling statement;

deleting said reference structure; and

freeing said memory block.

45. (Currently Amended) The method of claim ~~[[44]]~~ 38, wherein said memory structure is a stack.